L11: Entry 1 of 1 File: USPT May 7, 2002

DOCUMENT-IDENTIFIER: US 6385621 B1
TITLE: Computer software for maintenance resource management

Brief Summary Text (6):
Another software program available which is useful in resource planning is a job standards program used to create, maintain and manipulate standards for maintenance work, that is detailing the steps to be taken and manhours/trades needed to complete a task, allowing a user to construct new standards using information retained in a database. For example AutoMOST.TM., available from H.B. Maynard and Company, Inc. performs this function.

Brief Summary Text (19):
Utilizing the present invention, a user can use historical data and multi-activity analysis to identify the steps needed to complete a new job plan or use existing operation/suboperation job standards and integrate that data directly into the job plan. This substantially reduces the time needed to estimate the time, tools, material and labor needed to complete a job and optimizes resource utilization by developing a job plan based on job standards. Further, utilizing the capabilities of the job standards program increases the ability to accurately plan for manhour allocation, and material utilization, avoiding errors or omissions that can result in a poor plan that disrupts the overall manpower resource allocation.

Detailed Description Text (20):
Of course, other questions are possible such as identifying special tools required, safety steps required, such as equipment lock out which may impact setup time, etc. Typically, a database for a facility will have the specific equipment predefined so that drop down lists are available. Once a piece of equipment is selected, the question can be whether you wish to select an operation/suboperation for that equipment that was previously developed. That is, the operation for replacing the impeller could have been developed before, and if so, could be selected also from a drop down list, to utilize the existing standard specification. The job crew size, manhours/duration, etc., are calculated automatically after the job operation is defined. At that point, the user can override the specified parameters and request an updated calculation to see how the changes affect the job plan.

Detailed Description Paragraph Table (6):
Description: REPLACE BROKEN SHEAR PIN Assume: Schedule Factor = 2.0 One-way travel = 0.25 hours Daily Duration Prep 10 REPLACE BROKEN SHEAR PIN 1.62 0.00 IN WICKET GATE 1. Remove keeper plate 2. Setup hydraulic jack 3. Install jacking rod 4. Jack out shear pin 5. Hand ream shear pin hole 6. Install shear pin 7. Install keeper plate 8. Replace detector piping 20 REPLACE SHEAR PIN 3.46 0.22 1. Remove keeper plate 2. Setup hydraulic jack 3. Install jacking rod 4. Jack out shear pin 5. Hand ream shear pin hole 6. Install shear pin 7. Install keeper plate 999 TRAVEL 0.50 Total 5.58 hrs LABOR DISTRIBUTION OP CRAFT CODE QTY HOURS MANHOURS 10 MECHJP 2 1.62 3.24 20 MECHJP 2 3.46 6.92 999 MECHJP 2 0.50 1.00 TOTALS 5.58 11.16 WORK ORDER CALCULATIONS Step Op Description Calculation 1. 10 Duration after Schedule Factor 0.81 hrs * 2.0 = 1.62 hrs Daily Prep after Schedule Factor 0.00 hrs * 2.0 = 0.00 hrs Total 1.62 hrs 2. 20 Duration after Schedule Factor 1.62 hrs * 2.0 = 3.24 hrs

Daily Prep after Schedule Factor 0.11 hrs * 2.0 = 0.22 hrs Total 3.46 hrs 3. For application of job prep 10 Number of days 1.62 hrs/8.0 hrs - (2 * 0.25 hrs) - 0.00 hrs = 1.62 hrs/7.5 hrs = 0.22 = 1.0 day 20 Number of days 3.46 hrs/8.0 hrs - (2 * 0.25 hrs) - 0.22 hrs = 3.46 hrs/7.28 hrs = 0.48 = 1.0 day For application of total travel time (1.62 hrs + 3.46 hrs)/(8.0 hrs - (2 * 0.25 hrs) - (0.00 + 0.22) hrs = 5.08 hrs/7.28 hrs = 0.70 = 1.0 day 4. 10 New Duration 1.62 hrs + (1.0 day * 0.00 hrs) = 1.62 hrs 20 New Duration 3.24 hrs + (1.0 day * 0.22 hrs) = 3.46 hrs 5. 999 Travel Time 1.0 day * (2.0 * 0.25 hrs) = 0.50 hrs Note: The bold numbers are the ones used in the work order on the previous page.

CLAIMS:

3. The system of claim 1 wherein the first software program has access to a database containing data tables selected from the group consisting of equipment tables, labor tables, manhour tables, operation tables, material tables, tool tables and combinations thereof.

L20: Entry 1 of 7 File: USPT Dec 3, 2002

DOCUMENT-IDENTIFIER: US 6490569 B1
TITLE: System for combining life cycle assessment with activity based costing using a relational database software application

Brief Summary Text (13):
The software should also be network configurable for work-group applications and allow for report generation that is flexible to allow for selecting, rearranging, and graphically presenting results in a variety of ways. Information may have to be presented to senior management for strategic planning or manufacturing facility engineers for product/process improvement or even to a local citizen's action group (CAP) for review of environmental burden(s). The software must be coupled with the relational database and the database should be accessible independent of the rest of the software. Modules or segments should be hierarchical so that the tasks associated with any item can be easily modeled and the drivers and task elements changed as frequently as required.

Brief Summary Text (14):
Object-oriented programs manipulate objects rather than data or text per se. Objects represent collections of items (both data and the functions that operate on those data) and can have information associated with them through address pointers that refer to other objects. For example, a screen object such as a process box, can itself be an information icon on a screen, but it must be linked to other sets of information in the database. When the user "clicks" on the object with a mouse, the software automatically links the screen object with the related information. While such developed software programs have existed for conventional mass and energy balance LCA work, the need remains for software that combines these aforementioned features with the ability to perform simultaneous, sequential, or subsequent LCA/ABC analysis using task oriented drivers.

Brief Summary Text (26):
The LCAPIX module reduces the product, process, or service into 5 or more (or less) separate and distinct stages (including the product name) or levels as also shown in FIG. 7, and allows for completing ABC or LCA studies simultaneously (or individually at an earlier or later time). The relational database or library allows for simple access to stored data that may be associated with any of the levels or stages. Before the reduction of this invention to practice, no known practical hierarchical scheme generically describing a product, process, or service existed such that a useful associated software application combining LCA with ABC could be developed. By first perfecting this hierarchical scheme and subsequently developing the LCAPIX software application incorporating this scheme, LCA/ABC studies that heretofore required weeks or months can now be accomplished in days or even hours. Further, by development of the LCAPIX software module, instead of a group of individuals (or costly consultants) knowledgeable about LCA and ABC and associated engineering science and accounting principles, the requirement to complete an LCA/ABC study can be performed by one individual who is familiar with how the process, product, or service is accomplished. This is a feature that allows for large cost savings in both manhours (labor) and wages.

Drawing Description Text (12):
FIG. 10, is a flowchart describing how the LCAPIX module software works beginning

at the entry point, G, for the Process Entry point for the Inventory Stage (I) corresponding to level or stage 1. Beginning at the entry point G, the product and machine line names are retrieved from the database.

Drawing Description Text (13):
FIG. 11 is a flowchart describing how the LCAPIX module software works beginning at the entry point, I, for the Machine Line Entry Point for the Inventory Stage (I), corresponding to level or stage 2. Beginning at the entry point I, the machine line and subprocess names are retrieved from the database.

Drawing Description Text (14):
FIG. 12 is a flowchart describing how the LCAPIX module software works beginning at the entry point, J, for the Subprocess Entry Point of the Inventory Stage (I), corresponding to level or stage 3. Beginning at the entry point J, subprocess and operation data are retrieved form the database.

Drawing Description Text (15):
FIG. 13 is a flowchart describing how the LCAPIX module software works beginning at the entry point K, for the Operation Entry Point of the Inventory Stage (I), corresponding to level or stage 4. Beginning at the entry point K, the operation and task data are retrieved from the database. .

Drawing Description Text (16):
FIG. 14 is a flowchart describing how the LCAPIX module software works beginning at the entry point H, for the Baseline or Task Entry Point of the Inventory Stage (I), corresponding to level or stage 5. Beginning at the entry point H, the baseline process window is displayed and it must be decided if the baseline must be updated or not.

Drawing Description Text (17):
FIG. 15 is a flowchart describing how the LCAPIX module software works regarding the use of the Elemental Component entry point of the Inventory Stage (I). For the Elemental Component of the Inventory Stage, retrieval of the elemental component data from the database is first accomplished.

Drawing Description Text (18):
FIG. 16 is a flowchart describing how the LCAPIX module software works for the Elemental Component Driver Entry Point of the Inventory Stage (I). For the Elemental Component of the Inventory Stage (I), retrieval of the elemental component driver data is necessary. Next, the elemental component driver window is displayed. Then a decision is required regarding updating the elemental component driver information.

Drawing Description Text (34):
FIG. 32 is a Product Specification window which indicates the product title and displays the version of the model based on user input. Here a summary of previous work can be reviewed.

Drawing Description Text (44):
FIG. 42 is a block diagram showing the main hardware components of a computer system, or workstation, which may be used to implement the invetion. The invention is carried out using any computer or microprocessor system which includes standard computer components such as a hard disk drive, a floppy disk drive, an optical disk drive, a RAM, and a ROM for storing computer microcode.

Detailed Description Text (2):
Detailed descriptions of the preferred embodiment are provided herein. It is to be understood, however, that the present invention may be embodied in various forms. As software and hardware applications become ever more complex and faster, it is likely that the inventive concepts contained herewithin will be used to achieve

near or actual real time data access and manipulation from databases that are publicly accessed via the internet. The software application described herewithin could be developed to include Internet access and provided for workgroups in cyberspace. Therefore, specific details disclosed herein are not to be interpreted as limiting, but rather as a basis for the claims and as a representative basis for teaching one skilled in the art to employ the present invention in virtually any appropriately detailed system, structure or manner.

Detailed Description Text (18):
FIG. 18 is a flowchart for Associating Drivers to Load Substance Categories Entry Point of the Inventory Stage (I) at entry point, O, 1350. For the Associated Drivers of the Inventory Stage, a decision is made regarding updating the driver factors, 1400. If yes, then update the driver factors, 1410. Then the driver factors are saved 1420 and revert back to 1400 continuously until all necessary, desired, and sufficient driver factors have been updated. If no update is necessary, the decision becomes associating the drivers with load categories, 1430. If yes, make the association, 1440, drivers must be matched with load substance categories. The associations are saved, 1450, and automatic return to 1430 to associate more drivers until all drivers and load substance categories are associated. If no association, then automatic return to the Main Menu window, 330.

Detailed Description Text (28):
The foregoing explanation describes the logic flow of the software application, LCAPIX and the function it provides. This invention is carried out using any computer or microprocessor system which includes standard computer components such as a hard disk drive, a floppy disk drive, an optical disk drive, a RAM, and a ROM for storing computer microcode. Included herewith is a block diagram of a computer system, or workstation, which may be used to implement the invention shown in FIG. 42. Further the invention may be carried out using a plurality of computers which are networked together in a manner which would be used in comprehensive costing analysis approaches.

CLAIMS:

11. The computer-assisted method according to claim 8, further comprising the steps of: receiving first information sufficient to identify at least one product, process or service; and receiving second information reflective of a time frame in which said at least product, process or service is to be provided; wherein said first and second information is received via a graphic user interface.

First Hit Fwd Refs

Generate Collection Print

L20: Entry 3 of 7 File: USPT Jun 16, 1998

DOCUMENT-IDENTIFIER: US 5767848 A
TITLE: Development support system

Brief Summary Text (3):
A development support system for supporting the cooperative work of a plurality of persons is disclosed in, for example, Japanese Patent Laid-open (Kokai) No. 3-250365. This support system enables a plurality of members of a task group stationed at separate places to hold a conference at every stage of cooperative work using a plurality of information processors for exchanging information about their individual achievements and proceeding their work with the information obtained in the conference.

Brief Summary Text (4):
A development management system disclosed in Japanese Patent Laid-open (Kokai) No. 4-364529 enables a plurality of members of a software development task group to manage the software development activities by exchanging electronic mails containing information about the contents of module development, such as command codes, projects, work codes, work data, requesters, destinations of request and priority, and development management information. Since this development management system uses electronic mails, information about development activities can be sent to the members regardless of the members' circumstances, and the development management need not use any printed matters, such as documents.

Brief Summary Text (5):
Incidentally, when carrying out a development project planned to develop a new product, the general target value of the development project, the estimated progress of the development project, the target values of every task of the development work and the estimated progress of the divisions of the development work must be coordinated at stages of the development project.

Brief Summary Text (6):
When the aforesaid known development support system is employed to support the cooperative work and information is merely exchanged between the members, the individual members must manage the information for coordination and, consequently, load on the members, i.e., the users of the development support system, for management work to achieve the targets of the performance and the cost of the product and development schedule cannot be reduced satisfactorily. This known development support system is thus incapable of satisfactorily supporting management work for coordinating the general target or the individual targets of the development project and the expected progress of the development project.

Brief Summary Text (9):
A first object of the present invention is to provide a development support system capable of supporting management work on coordinating the general target and individual targets of a new product development project and estimated data representing the progress of the new product development project.

Brief Summary Text (32):
When the target values are changed or particularized, the general target value and the individual target values are examined to see whether or not the general target

value and the individual target values match with each other and, at least when the general target value and the individual target values do not match with each other, a notice is given to the members of the development project team participating in the development activities to that effect.

Brief Summary Text (33):
Accordingly, each member of the development project team is able to particularize a rough development plan made at the initial stage of development activities during the progress of development activities, to confirm the coordination of the estimated values of progress of general and individual activities objectively determined on the basis of design data and target values and the coordination of the target values of general activities and those of individual activities, and to work out an optimum plan to achieve the object.

Brief Summary Text (35):
Since the information about the progress management parameters are not obtained by exchanging electronic mails for development management, development management does not require any special work. Since the information is extracted from electronic mails relating to procedures essential to the development activities, the information is not different from the actual state of development activities, and the condition of the progress of development activities can be objectively monitored without requiring any additional work.

Drawing Description Text (17):
FIGS. 16(1) and 16(2) are a table and a graphic representation, respectively, of progress management information displayed on screens, showing the condition of progress;

Drawing Description Text (26):
FIGS. 25(1) and 25(2) are a table and a graphic representation displayed on screens, respectively, showing the schedules and the actual conditions of progress of tasks;

Detailed Description Text (4):
The server machine 2000 has a cost estimating unit 5 for estimating costs on the basis of the product models linked by the development activity models, a performance estimating unit 6 for estimating the performance on the basis of the product models linked by the development activity models, a schedule estimating unit 7 for estimating development schedule on the basis of the amount of work and resources for each unit task of the development activity models, and an estimate storage unit 8 for storing estimates provided by the units 5, 6 and 7. The estimates are stored in an estimated cost storage unit 81, an estimated performance storage unit 82 and an estimated schedule storage unit 83, respectively.

Detailed Description Text (9):
Then, as shown in FIG. 6, electrical designers and the like, for example, design electrical parts assigned to the electrical designers, store the design data of the electrical parts in the product model storage unit 2, estimate the amount of work necessary for completing development activities which are in the course of design and not started yet, and set activity programs in the development activity models as indicated at 601. Members in charge of experimental model manufacturing and testing, as well as designers, store the results of work to carry out tasks assigned thereto and included in the development activity models in the product model storage unit 2, estimate the amount of work necessary for completing tasks which are in the course of execution and not started yet, set activity programs in the development activity models, and store resource data including facilities and personnel available for development activities in the resource model storage unit 4 as indicated at 602.

Detailed Description Text (10):

The schedule estimating unit 7 estimates the condition of progress of the development project at the current time and the amount of work necessary for carrying out the rest of the development tasks on the basis of the development activity models as indicated at 603, makes an estimated development schedule by using the resource data including available facilities and available personnel stored in the resource model storage unit 4, and stores the estimated development schedule in the estimate storage unit 8 as indicated at 604.

Detailed Description Text (11):
A calculation procedure to be carried out by the schedule estimating unit 7 will be described hereinafter with reference to a flow chart shown in FIG. 8. In step 801 current schedule data representing the present state of the development project is fetched from the development activity model storage unit 3. Then, in step 802, information about the resources is fetched from the resource model storage unit 4. Then, the degree of progress of the development project is calculated in step 804, and the remaining manhours are estimated in step 805 by using definitions of the degree of progress of the development project shown in FIG. 9. Two indices shown in FIG. 9 indicate estimated remaining manhours obtained through the statistical analysis of completed tasks. The index 1 is used for estimating the remaining manhours on the basis of the number of tasks which have been finished before the time of estimation and manhours spent for finishing those finished tasks. The index 2 is used for estimating remaining manhours on the basis of problems which have been finished before the time of estimation and manhours spent for solving the problems. The problems are those obstructing the progress of the development project, such as troubles occurred in experimental models during tests. Expression showing the index 1 has a singular point at the start of the development project, wherein the denominator is zero, and expression showing the index 1 has a singular point when the number of solved problems is zero. These singular points are avoided to estimate the general schedule of the development project.

Detailed Description Text (19):
The function of a copy/move button 1110 will be described hereinafter. The precedence relation between the tasks is determined beforehand. The precedence relation is specified by operating the model creating units 1 of the client machines 1000 at the early stage of planning. The precedence relation defines restrictive conditions, such as "A task B should not be executed unless a task A becomes acceptable." Copying and moving of the tasks defined by the precedence relation will be described with reference to FIG. 12. Suppose that the order of execution of tasks B and C are defined by the precedence relation and the task C cannot be executed unless the task B becomes acceptable. First, "Copying" will be described. FIG. 12(a) shows a state where the task B has not become acceptable and is to be executed again two days later. In this development support system, the task B can be copied by operating a copy button to copy relevant test items. When the copy button is operated, all the task attributes except the date of execution are copied. When it is specified to execute the task B two days later after copying the task B, the task C is moved automatically to a position next to that of the task B as shown in the right section of FIG. 12(a). Unoccupied days are occupied automatically by tasks following the task C and the schedules of tasks following the task C are shifted automatically backward. "Moving" will be described hereinafter. FIG. 12(b) shows a state where the task B could not be executed on scheduled days for some causes and the execution of the task B is delayed forcibly two days. When the execution of the task B is delayed two days, the task C in the precedence relation with the task B is moved automatically, and tasks D and E which have been behind the tasks B and C are advanced automatically to unoccupied days from which the tasks B and C have been moved, the consequence of which is shown in the right section of FIG. 12(b). The plan for the execution of the tasks can be quickly worked out infallibly because the order of execution of the tasks is defined by the precedence relation.

Detailed Description Text (20):

The operations of the communication management unit 11 based on the conference between the members of the development project team for adjusting the target values stored in the target value storage unit 12 and giving inconsistency alarms will be described by way of example with reference to FIG. 13. When a problem that will impede the progress of the development project arises in the course of the development project, countermeasures must be taken immediately so that the schedule of the development project may not be delayed. However, the schedule is framed and amended by the testing department, while the countermeasures are worked out by the design department, and the countermeasures are executed and the experimental model is modified by the testing department. Accordingly, the testing department must know the schedules of the design department and the experimental model testing department to determine a date for retesting in case a problem arises in the experimental model. The communication management unit 11 is used for acquiring information about the schedules of other departments and for setting a term on the basis of the schedules of other departments. FIG. 13 shows a table of assistance in explaining a schedule input operation for making an agreement on a task 4. When the task 4 shown on a schedule table shown in FIG. 11, a window is opened in the schedule table, and a problem section 1301 in which a problem is mentioned and an action section 1302 in which a countermeasure for the problem is mentioned are displayed. The members enter promissory schedules in sections for the departments to which they belong. For example, the member of the design department enters the promissory schedules in a sections 1303 for design change schedule, a section 1304 for design change completion and a section 1305 for desired modification schedule. The member of the experimental model manufacturing department enters promissory schedules in a section 1306 for modification schedule, a section 1307 for modification completion and a section 1308 for desired test schedule. The testing department that works out a test schedule is able to determine a new test schedule with reference to the promissory schedules indicated in the window. This development support system enables making reference to the data entered in the table by the member in charge even if the relevant member is away when an inquiry is made by phone. The development support system monitors the schedules entered in the section 1303 for design change schedule and the section 1304 for design change completion schedule by the design department continuously. If any date is not entered in the section 1304 for design change completion date after the date entered in the section 1303 for design change schedule, the development support system gives an alarm and blinks a red light indicating the design department. An alarm is given also when the dates entered in the section 1306 for modification schedule and the section 1307 for modification completion relating with the experimental model manufacturing department are delayed. The development support system manages time limits automatically and gives alarms in this way when the scheduled dates are delayed to limit the delay of the schedules to the least possible extent.

Detailed Description Text (25):
As the designing work progresses and the estimated cost of each unit for which the target cost is determined is determined, and the estimated cost is greater than the target cost, the estimated value and target value reviewing unit 9 inform the members to that effect and requests the members to take measures for reducing the estimated cost as indicated at 405. A method of changing the design of parts of estimated costs exceeding the corresponding target costs to reduce the estimated costs and a method of adjusting the target costs of such parts to the estimated costs are possible measures. However, if the latter method is employed, the sum of the target costs of the parts does not coincide with the target cost of the product. In such a case, the general/individual target value consistency maintaining unit 10 detects the discrepancy between the sum of the target costs of the parts and the target cost of the product, informs the members of the development project team to that effect and requests measures for adjusting the discrepancy. Possible measures are a method of making the sum of the target costs of the parts coincide with the target cost of the product by reducing the target costs of other parts and a method of making the sum of the target costs of the

parts coincide with the target cost of the product by changing the target cost of the product. The selection of either method decided by a conference between the members as indicated at 407 or by a responsible person participating in the development project. The communication management unit 11 stores the conclusions of the conference in the target value storage unit 12 as indicated at 406.

Detailed Description Text (34):
As is apparent from the foregoing description, the development support system in this embodiment stores the product models expressing the structure and the characteristics of a product comprising component parts designed respectively by the members assigned thereto in the product model storage unit, the performance and the cost of the product is estimated on the basis of the product models, estimated schedules are determined on the basis of the development resources and the amounts of development work necessary for developing the component parts, and an estimated general schedule of the product development project is determined by adding up the estimated schedules for developing the component parts. The members of the development project team are thus able to use in common the estimated general schedule of the development project that varies as the development project progresses. The estimated general schedule and the estimated individual schedules are compared with the corresponding target schedules, and the development activity program can be adjusted so that the target schedules can be achieved. The targets and the plans of the mutually related activities can be automatically corrected or improved on the basis of information contained in the inquiries and replies exchanged between the members of the development project team to update the target values continually and the members are able to access the updated target values. Accordingly, the members participating in the product development project particularize the rough development program worked out in the early stage of the product development project as the product development project progresses to maintain the consistency of the estimated general schedule and to work out an optimum program to achieve the target values.

Detailed Description Text (41):
The members of the development project team uses, for example, CAD tools (computer-assisted design tools) and terminal information processors to carry out the work concerning the development project and to perform administrative communications necessary for carrying out their own work, such as requesting estimation, replying to the request for estimation, placing orders, reporting delivery, giving instructions for trial manufacture and communicating the achievement of manufacture. The members participating in the development project are not limited only to those whose principal work is activities for the development of the product but may include the personnel of other management departments. This development system uses standard electronic mails for administrative communications for communicating information about the progress management items, among the administrative communications. The development tools 101, 102, 103, . . . need not necessarily be connected to the information network 104, standard electronic mail input/output terminal devices may be connected to the information network 104 instead of the development tools.

Detailed Description Text (45):
The foregoing procedure enables the members of the development project team to grasp the actual state of progress of development of the modules and the parts of the product to be developed, namely, 1) when the plotting work was started, 2) when the drawings were completed, 3) when estimation was requested, 4) when the estimation was given, 5) when orders were placed for materials, 6) when the materials were delivered, and 7) when an instruction to manufacture an experimental model was given and 8) when the experimental model was manufactured.

Detailed Description Text (46):
FIGS. 16(1) and 16(2) illustrate the progress management information stored in the progress management information storage unit 1404 to be displayed by the progress

management information reference unit 1405. FIG. 16(1) is a representation in a tabular form and FIG. 16(2) is a representation in a graphic form.

Detailed Description Text (48):
In FIG. 16(1), the date 1603 of request for the estimation of the cost of a cabinet 1602, which is one of the component parts of the objective product, is January 15, and any reply 1064 to the estimation request has not yet been made. In FIG. 16(2) showing the pieces 1611 to 1618 of the progress management information in a graphic form, special marks 1621 to 1628 represented by colors, patterns or combinations of colors and patterns are assigned to the pieces 1611 to 1618 of the progress management information to indicate the completion of processes corresponding to the pieces 1611 to 1618 of the progress management information. The progress management items are the direct indices of the progress of tasks for the development of the product. For example, in FIG. 16(1), the determination of the date of every progress management item indicates one step of progress of the task for the development of the product. Hereinafter, a process corresponding to each progress management item will be referred to as a progress step.

Detailed Description Text (49):
Shown in a graphic form on the output screen 1600 are progress steps 1611 to 1618 (processes corresponding to the progress management items), the marks 1621 to 1628 respectively corresponding to the progress steps 1611 to 1618, and the conditions of progress of preparation of the modules and parts at the nodes of the product configuration tree, indicated by the marks. Data numerically indicating the condition of progress in FIG. 16(1) are indicated by the predetermined marks in FIG. 16(2). Although the representation in a graphic form is unable to indicate the dates exactly, the graphic representation has an advantage that the general condition of progress can be readily and visually recognized, for example, when the darkness of the color, i.e., the mark, is increased as the development activity progresses. Although the processes performed at the latest times are indicated in this embodiment, all the processes which have been performed up to the present may be indicated by marks.

Detailed Description Text (62):
After the estimated dates have been set by the progress management information setting unit 1407, the progress management information reference unit 1405 is started to display comparatively both the estimated dates of completion of the tasks corresponding to the progress management items and actual dates of completion stored in the progress management information storage unit 1404 to enable the comparison of the estimated progress and the actual progress of the tasks. The displaying operation of the progress management information reference unit 1405 can be started by clicking a progress reference button 2402 provided on the screen 1600 for setting estimated dates shown in FIG. 24. The information may be displayed on the screen in either a tabular form numerically showing both the estimated dates and the actual dates in combination as shown in FIG. 25(1) or a graphic form indicating the actual progress of the tasks by predetermined marks as shown in FIG. 25(2). Either the tabular form of FIG. 25(1) or the graphic form of FIG. 25(2) can be selected by clicking either of a date button 2503 and an alarm button 2504 which are displayed on the screen 1600 when a progress reference button 2402 displayed on the screen 1600 is clicked. When the tabular form is selected, the estimated dates 1101 of completion and actual dates of completion of processes 1611 to 1618 for the progress management items are displayed in combination as shown in FIG. 25(1). When the graphic form is selected, processes corresponding to the progress management items behind the schedules and processes corresponding to progress management items in accordance with the schedules are marked with marks 2505 and 2506 different from each other in color or shape so that those processes can be clearly identified. The graphic form of representation of the information by this embodiment, and the representation with color marks or pattern marks assigned to the progress management items as shown in FIG. 16(2) may be used in combination to specify the processes behind the schedules concretely. Since the estimated completion dates set

for the processes corresponding to the progress management items and the corresponding actual dates of completion can be compared, the condition of progress of the development project can be more exactly recognized.

Detailed Description Text (65):
When a problem arises, it is thus unnecessary consider to which measured data reference should be made to look into the causes of the problem when the table showing the categories of problems in connection with the corresponding measuring items is used and, therefore, work on looking into the causes of the problem can be very efficiently carried out. Since the measured data can be obtained in a real time mode, measures for solving the problem can be promptly conceived and taken.

Detailed Description Text (68):
The present invention solves the problem in the conventional development support system that special work for development management is necessary and the management of the development project is a burden on the members participating in the development project.

CLAIMS:

1. A development support system for supporting product development activities of a plurality of members participating in a product development project for developing a new product, comprising:

model storage means for storing product models, resource models to be used for product development, and product development activity models;

target storage means for storing general and individual target values for the cost and the performance of said product, and development schedules;

estimating means for estimating the cost, the performance, and the progress of development schedules on the basis of models stored in the model storage means;

reference support means for supporting members of a development project team to make reference to the stored models and target values and the estimated data of progress;

model change and particularization support means for supporting the members of the development project team for operations to change and particularize the stored models; and

notifying means for deciding whether or not the estimated data of progress meet the corresponding target values when the models are changed or particularized and, at least when the estimated data of progress do not meet the corresponding target values, notifying the members to the effect that the estimated data of progress do not meet the corresponding target values,

wherein each product development activity model is a graphic representation describing, in connection with a precedence relation between development activity units, a name for each development activity unit, restrictions on each development activity unit, the resource models to be used by each development activity unit, the product models to which each development activity unit makes reference, and product models showing the results of each development activity unit.

12. A development support system for supporting product development activities of a plurality of members of a development project team, comprising:

model storage means for storing product models, resource models to be used for product development, and product development activity models;

target storage means for storing general and individual target values for the cost and the performance of said product, and development schedules;

estimating means for estimating the cost, the performance, and the progress of development schedules on the basis of models stored in the model storage means;

reference support means for supporting members of the development project team to make reference to the stored models and target values and the estimated data of progress;

model change and particularization support means for supporting the members of the development project team for operations to change and particularize the stored models;

target change and particularization support means for supporting the members of the development project team for operations to change and particularize the target values; and

notifying means for evaluating the consistency of the general target value and the individual target values with each other when the target values are changed or particularized and, at least when the general target value and the individual target values are inconsistent with each other, notifying the members to that effect,

wherein each product development activity model is a graphic representation describing, in connection with a precedence relation between development activity units, a name for each development activity unit, restrictions on each development activity unit, the resource models to be used by each development activity unit, the product models to which each development activity unit makes reference, and product models showing the results of each development activity unit.

23. A development support system comprising:

a plurality of terminals; and

analysis equipment connected to said plurality of terminals, said analysis equipment having first storage means for storing work information of a product output from said plurality of terminals, second storage means for storing of target values for at least one cost and development schedules of the product, estimating means for estimating at least one of cost and development schedules of the product based on the work information, judging means for judging whether or not the at least one of estimated cost and development schedules of the product estimated by said estimating means meet, respectively, said target values for at least one of the cost and development schedules of the product, and notifying means for notifying the terminals when at least one of the estimated cost and development schedules does not meet, respectively, said target values for at least one of the cost and development schedules of the product.

24. A development support system according to claim 23, wherein said work information is a work quantity about a changed part of the product, and

said estimating means estimates the development schedules of the product based on the work quantity.

25. A development support system according to claim 23, wherein said work information includes information concerning components about a changed part of the product, and

said estimating means estimates the cost of the product based on the components.